

NASA Technical Memorandum 106618
ICOMP-94-10; AIAA-94-3209

1N-07
12641
13P

Computational Methods for HSCT-Inlet Controls/CFD Interdisciplinary Research

Gary L. Cole and Kevin J. Melcher
Lewis Research Center
Cleveland, Ohio

Amy K. Chicatelli and Tom T. Hartley
University of Akron
Akron, Ohio

and

Joongkee Chung
Institute for Computational Mechanics in Propulsion
Lewis Research Center
Cleveland, Ohio

Prepared for the
30th Joint Propulsion Conference
cosponsored by AIAA, ASME, SAE, and ASEE
Indianapolis, Indiana, June 27-29, 1994



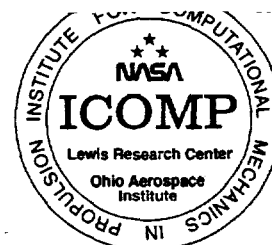
National Aeronautics and
Space Administration

N94-35352

Unclas

G3/07 0012641

(NASA-TM-106618) COMPUTATIONAL
METHODS FOR HSCT-INLET CONTROLS/CFD
INTERDISCIPLINARY RESEARCH (NASA.
Lewis Research Center) 13 p



COMPUTATIONAL METHODS FOR HSCT-INLET CONTROLS/CFD INTERDISCIPLINARY RESEARCH

Gary L. Cole and Kevin J. Melcher
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

Amy K. Chicatelli* and Tom T. Hartley†
University of Akron
Akron, Ohio 44325

and

Joongkee Chung‡
Institute for Computational Mechanics in Propulsion
Lewis Research Center
Cleveland, Ohio 44135

Abstract

A program aimed at facilitating the use of computational fluid dynamics (CFD) simulations by the controls discipline is presented. The objective is to reduce the development time and cost for propulsion system controls by using CFD simulations to obtain high-fidelity system models for control design and as numerical test beds for control system testing and validation. An interdisciplinary team has been formed to develop analytical and computational tools in three discipline areas—controls, CFD, and computational technology. The controls effort has focused on specifying requirements for an interface between the controls specialist and CFD simulations and a new method for extracting linear, reduced-order control models from CFD simulations. Existing CFD codes are being modified to permit time accurate execution and provide realistic boundary conditions for controls studies. Parallel processing and distributed computing techniques, along with existing system integration software, are being used to reduce CFD execution times and to support the development of an integrated analysis/design system. This paper describes: the initial application for the technology being developed, the high speed civil transport (HSCT) inlet control problem; activities being pursued in each discipline area; and a prototype analysis/design system in place for interactive operation and visualization of a time-accurate HSCT-inlet simulation.

Introduction

The design of propulsion controls for complex vehicles, such as the proposed high-speed civil transport (HSCT), relies on accurate models of the system components. These models can be derived from experimental data or computer simulations. Simulations are more desirable because they can reduce development time and cost associated with experimental testing. Two major categories of simulations exist—based either on large lumping techniques or computational fluid dynamics (CFD).¹ Traditionally, the controls specialist has relied on the former to generate low-order linear models convenient for control system design. A major advantage of lumped-parameter simulation is the rapid execution time, including real-time operation for in-the-loop hardware testing. However, for high-speed flow, such as in the Mach 2+ regime of the HSCT, lumped parameter models fail to capture much of the relevant flow physics, which could result in a suboptimal control design. This ultimately translates into increased development time and cost associated with tuning the control system via experimental means.

An alternate approach would be to resort to CFD simulations. CFD provides a much more accurate representation of internal flow phenomena, including shock waves and viscous boundary layers. However, high-level CFD simulations (i.e., multidimensional,

*Research Associate, Student Member AIAA

†Professor, Electrical Engineering, Member AIAA

‡Research Associate, Member AIAA

Copyright ©1994 by the American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for government purposes. All other rights are reserved by the copyright owner.

Euler/Navier-Stokes equations) are seldom, if ever, used.² This is especially true for controls specialists because CFD simulations: 1) are not interactive nor easily operated by non-CFD experts; 2) have relatively long execution times that are impractical for controls problems; 3) can have millions of states for which no suitable/robust model-reduction techniques are available; and 4) generally lack moving geometry surfaces. Reduced development time for propulsion system controls clearly should be achievable through the use of high-fidelity system models and testing of control designs via "numerical experiments" based on CFD. The question is how to make CFD more practical for use by the controls discipline. To address the issues enumerated above, the NASA Lewis Research Center has initiated a controls/CFD cross discipline research project. An interdisciplinary team has been formed to pursue research and development activities in three discipline areas—controls, CFD, and computational technology. This paper begins by describing the HSCT inlet control application, thus setting the stage for an overview of the activities in each discipline area. A major focus of the paper is on the computational technologies being brought to bear on the development of an integrated analysis/design system prototype. Interactive execution and visualization of a time-accurate HSCT inlet simulation is used to demonstrate the initial operating capabilities of the analysis/design system. The paper concludes with a summary of progress and some remarks about possible directions for future research.

HSCT Inlet Control Application

Because of its relevance to NASA's high speed research program, the HSCT inlet control problem was selected as the first application for the controls/CFD research. Issues of efficiency, noise, and emissions make the propulsion system a key factor in the development of an economically viable HSCT. The propulsion system must deliver high performance while maintaining the desired operability. For the inlet this means delivery of high pressure-recovery, low-distortion air to the engine, with a safe margin of inlet stability. A schematic of the inlet control is shown in figure 1. The inlet is a mixed-compression type, having supersonic compression occurring both internally and externally. An HSCT-baseline inlet configuration (axisymmetric with a translating centerbody) is being used for the current controls/CFD research. Higher inlet performance is achieved as the normal shock is positioned closer to but downstream of the inlet throat, and the throat Mach number is decreased to (slightly above) 1.0. Unfortunately, as

inlet performance increases, stability margin decreases. Perturbations in free-stream conditions can cause the throat to choke or the shock to move forward of the throat. Engine transients can also cause the shock to move forward. Either case can result in a phenomenon known as inlet unstart during which the normal shock is expelled from the inlet accompanied by an abrupt decrease in pressure recovery and mass flow. The corresponding loss in engine thrust could severely hinder control of the aircraft causing considerable discomfort to the passengers. It is the job of the inlet control to maintain high performance conditions and to minimize or eliminate unstart occurrences.

The inlet control has two major control loops, as shown in figure 1. One regulates normal shock position in which the overboard bypass door position (area) is the primary manipulated variable. The other loop manipulates centerbody position (throat area) to regulate throat Mach number. (Centerbody position can also effect shock position.) The control design depends on knowledge of both steady and unsteady inlet flow-field behavior, such as the response of the normal shock position to perturbations in inlet geometry, free-stream flow conditions, and engine transients. This information is needed for many operating points throughout the flight envelope. The inlet control may be part of an integrated propulsion system and may also have to interact with the flight control system.

Controls Activities

Initially, the controls activity has two major thrusts. First, treating CFD simulations as "numerical experiments" will make it easier for the control engineer to effectively use CFD results for control design. This can be accomplished by building an interface that provides the "handles" to facilitate operation of the simulation and to extract the required information. This interface, which is part of an integrated analysis/design system described in a later section, is of prime importance. Establishing the requirements for such an interface was the first priority.¹ The second major activity focused on linear modeling and model reduction techniques for a one-dimensional CFD simulation.¹ This section briefly summarizes those activities.

The interface requirements were defined primarily from a controls point of view but involved considerable discussion among participants from all three disciplines. Figure 2 illustrates the major categories of simulation options required by the controls engineer.

The options consist of pre/post-simulation and interactive functions. The interactive functions permit the simulation to be operated as a "numerical experiment" by allowing run-time perturbations of boundary conditions and manipulation of moving geometry features. At the same time, the user would have the ability to select and view various simulation responses in near-real time. Pre-simulation options include items such as algorithm selection, numerical requirements of the code, and definition of initial conditions. Post-simulation options primarily permit off-line generation of information from CFD results and data analysis for the control engineer. Typical functions include display of the pre-simulation problem setup, requested output variables, and data analysis, such as frequency responses. The approach for implementing the required interface features, including an example for the HSCT inlet, is given in the "Computational Technology Activities" section.

The direct use of one-dimensional CFD methods to obtain control models had been performed previously³ and was selected as the initial approach for linear modeling and model reduction. The technique has been extended, as part of the controls/CFD program, to include the determination of uncertainty bounds which are necessary for robust-control specialists. The LAPIN quasi-one-dimensional, unsteady code⁴ was selected as the baseline CFD code to simulate an HSCT inlet. LAPIN steady-state data were used to create full and reduced order linear models of 123 and eight states, respectively. No discernible difference was found between the full and reduced order models. A comparison of the reduced-order models with transient data from LAPIN also showed excellent agreement.¹

A major advantage of the above linearization technique is that only spatial step information and steady-state operating conditions are required from the CFD simulation. However, the linear model is valid only near an operating condition. Hence, it is necessary to develop models for a number of operating points throughout the vehicle's flight regime. The corresponding steady-state CFD data are needed for each operating point and some transient data are also needed to validate the linear models. If the technique is extended to two and three-dimensional CFD simulations, as anticipated for the controls/CFD program, the number of states will increase significantly. This will have a dramatic impact on computing requirements for both the CFD and the model reduction process. The following numbers were extrapolated from sample calculations. At one megaflop per second, it will take approximately one minute to calculate a steady-state

operating point for LAPIN and nine minutes to do the model reduction from 132 states to eight states. A three-dimensional, full Navier-Stokes calculation with 500,000 grid points will take seven hours at 150 megaflops, and the model reduction, from 2,500,000 states to a manageable reduced-order model, will take an hour on a one teraflop machine. Hopefully, advances in computer technology and the application of distributed/parallel processing will result in more practical computing times for the multidimensional cases.

Computational Fluid Dynamics Activities

The LAPIN code was developed with many features that make it ideal for supersonic inlet controls applications. However, the intent is to make use of two and three-dimensional CFD codes with viscous capabilities. In addition the code must be suitable for use with control problems. Hence, there is a need for time accurate operation, realistic boundary conditions, and reasonable turn around times. Since we are not aware of such a multidimensional code, it was decided to modify an existing code rather than start anew. The PARC code⁵ was selected due to its fairly wide use in industry and government. Steady and transient flows based on the Navier-Stokes equations can be simulated using PARC. Either viscous or inviscid flow-field calculations can be made. PARC is available in two-dimensional/axisymmetric (PARC2D) and fully three-dimensional (PARC3D) versions. Initially, modifications were made to provide time-accurate operation and validation studies were conducted.⁶ Capabilities have been added to both versions of PARC which allow: specification of upstream and downstream (inlet exit) boundary conditions as a function of time, specification of compressor face average Mach number, which is equivalent to specifying corrected mass flow rate, and calculation of bleed mass flow rates based on local conditions and flow coefficients. The correct inlet exit or compressor face boundary condition is of particular importance.² The engine impedance can have a profound effect on the response of the inlet normal shock, as will be demonstrated in the "HSCT Inlet Application" section. It remains to be seen whether or not the exit boundary condition can be adequately specified in lieu of coupling the inlet simulation to a complete engine simulation.

Example unsteady calculations, using PARC, have been made for a variable-diameter-centerbody (VDC), mixed-compression inlet. Figure 3 shows results at several instants of time during a simulated inlet unstart transient, using inviscid PARC2D. (The multiblock

grid used to make the calculations is also shown.) In this case the free-stream static temperature was increased by three percent to induce the unstart. It took 7000 iterations to reach the initial steady-state condition (time $T=0.0$ sec). It took an additional 1900 iterations to reach the final unstarted state at 0.008532 second. The total CPU time for the 8900 iterations was approximately 300 seconds on one processor of a Cray Y-MP. This is somewhat lengthy for visualization in "real" time.

In an effort to provide more practical execution times for controls studies, the PARC code is being parallelized. Parallelization of PARC2D is being undertaken as part of a grant with Indiana University Purdue University at Indianapolis (IUPUI).⁷ Parallelization is accomplished by decomposing the grid into a number of blocks. The blocks are distributed to different processors to be solved in parallel. Blocks exchange information through block interfaces after each time step. Parallelization is achieved by using the parallel processing database GPAR, which was developed at IUPUI and is based on the APPL message passing paradigm.⁸ Preliminary timing results for the inviscid code have been obtained on the NASA Lewis cluster of IBM RS6000 workstations (LACE) interconnected by an ethernet network. The VDC inlet geometry was used with a 263×41 grid. The number of CPU seconds per time step for two, four, and five processors was 0.286, 0.147, and 0.131, respectively.⁹ A speedup of nearly two was achieved by going from two to four processors. Adding a fifth processor achieved a speedup of 2.2 rather than 2.5, as expected, relative to two processors. The less than linear speedup is attributed largely to the amount of communication time required as the grid is partitioned into more and more pieces. The CPU seconds per time step for the same case calculated on a single CRAY Y-MP processor was 0.056 or about 43% of the time required by five RS6000s.

PARC3D is being parallelized in-house at NASA Lewis. The approach is similar to that being used with PARC2D. Although the GPAR structure is not used, APPL is used for message passing. Preliminary timing results have been obtained on LACE for the VDC inlet problem. For the inviscid 3-D case a multiblock grid of seven blocks totaling 217,000 points was used. The elapsed seconds per time step for one and seven processors was 28.2 and 7.4, respectively.¹⁰ By combining three blocks onto one processor, it was possible to achieve better load balancing with five processors. The timing result was the same as for seven processors.¹⁰ The speedup relative to one

processor is 3.8 with a corresponding efficiency of 76 percent. The elapsed seconds per time step on a single Cray Y-MP processor was 4.0¹⁰ or about 54 percent of the time required by five RS6000s.

Results from these parallel programs are encouraging. Further optimization of the code should help to make massively parallel computing more efficient. That, combined with increasingly powerful processors, will hopefully achieve the desired result of making CFD codes, such as PARC, practical for use with controls-type problems.

At this time, all CFD computations using the PARC code were done for the VDC inlet. A geometry specification has been received for the HSCT-baseline, axisymmetric inlet. Generation of two and three-dimensional grids for use with PARC2D/PARC3D is underway.

Computational Technology Activities

A major task of the controls/CFD program is to develop an integrated analysis/design system that will make CFD simulation practical for use in the propulsion control design process. Such an environment will require the integration of a variety of controls and CFD software tools and support for the user interface functions described in the "Controls Activities" section (figure 2). Implementation of these requirements implies a need for the following computational concepts: information management (database inquiry and manipulation); libraries of dedicated service routines for visualization, searching, documentation, and data processing; and parallel/distributed processing. Figure 4 shows one possible arrangement for such features. The features encompassed by the solid-outlined boxes in figure 4 are provided by software being developed at NASA Lewis as part of the Integrated CFD and Experiment (ICE) project.¹¹ ICE is composed of three major subsystems which support: 1) experiment management and real-time data monitoring/acquisition, 2) CFD simulation, and 3) analysis (visualization and comparisons) of experimental and CFD data. Initial development of the ICE system has emphasized the experimental and analysis subsystems and focused on support of a multistage compressor flow physics program.

An explanation of the analysis/design system features, including enhancements/modifications required to interact with a CFD simulation, is given next. That is followed by an example using an HSCT inlet simulation as an application.

Integrated Analysis/Design System

The knowledge base shown at the top of figure 4 provides the foundation for the system. Configuration/Grids represents information about the article being "tested", such as a definition of the inlet geometry and bleed/bypass configurations. Parameters/Arguments/Results represents information about the CFD simulation—numerical method and time-step size, data request information, such as transient type and flow field variables, and CFD results files. Finally, notes documenting the simulation run, pictures of interesting results, etc. are included. The knowledge base holds information in a hierarchical fashion which links the various levels of information. This approach facilitates searches for, and retrieval of, information. A text file is used to define the knowledge base, making it easy to configure a system for specific test facilities and computational environments.

An extensive ICE code library provides a wide range of services which interface a "facility" (such as a CFD code) and the user with the knowledge base. Typical user interface and knowledge base management services, shown in the lower box, include visualization, search and documentation features. Services are provided by multiple dedicated processes (implemented in UNIX) that communicate through shared memory. Processes are managed by the ICE operating system. Process templates simplify the development of application-specific services (e.g. a run-time monitor) using a general structure from which reusable process management and user interface tools are derived. Since ICE is a multiprocess operating system, it can benefit from parallel processing when run on a shared-memory, multiprocessor machine.

Additional utilities were developed to simplify the interfacing of CFD simulation and controls analysis tools to the ICE environment. For example, some custom processes are required to interact with a CFD simulation. A process was developed to control and monitor the simulation and to acquire results based on the standard ICE run-time process template. This process sets up a shared-memory block that is used to exchange information with the CFD simulation. In order to permit interaction with distributed/parallel CFD simulations, the portable APPL message passing library is used. The ICE run-time process initiates execution of the APPL "compute" process which in turn starts and monitors the simulation and an APPL-ICE interface process. The interface process communicates with ICE via the shared-memory block and exchanges information with the CFD simulation via send and receive messages. Obviously, the CFD

simulation must be modified to include the proper send and receive messages. A multiprocess (parallel) simulation would have a master process to perform the information exchange with the interface process.

A similar capability, to interface directly to commercial control analysis tools (lower right of fig. 4), is not yet available. The initial approach will be to invoke an ICE process for processing CFD results attached to the knowledge base. The user will have the option of creating file(s) of flow field variables with data structure(s) required for the control analysis tool(s). These files would also be attached to the knowledge base. Eventually, the analysis tools would be invoked directly as part of the integrated analysis/design system. Other processes, such as ways for specifying certain types of boundary perturbations (e.g. frequency responses), are also desired. So far, making use of the ICE software for the controls/CFD program has been beneficial to both projects, because it reduces development time for the analysis/design system and provides feedback for the ICE development team.

HSCT Inlet Application

A prototype integrated analysis/design system has been developed for interactive execution and monitoring of a time-accurate simulation of the HSCT-baseline inlet. The LAPIN code, modified for message passing, was used as the CFD simulation. The run-time and interface processes, described above, were customized for use with LAPIN. Replacing the LAPIN code with the PARC 2D/3D codes should be a straight forward process.

A knowledge base was created to define the HSCT inlet configuration, the LAPIN CFD code, and the run-time monitor process. Figure 5 shows knowledge base records that are displayed when the ICE operating system is started. The left half of the screen shows major categories of records that define the inlet geometry and bleed/bypass configuration, specification of free-stream conditions, CFD related parameters, etc. A particular record can be selected for expansion to submembers by using a mouse to click on the button to its left. A check mark indicates the item selected (i.e. cfd-param[1]), and the expanded record is displayed on the right. It shows items related to the numerical algorithm (split characteristics), time-step size, etc. A more detailed description of an individual member can be displayed by clicking on its button, followed by clicking the question mark at the top of the screen. A display pops up with information, such as min/max values, alarm values, and special requirements. This latter feature assists the user, such as a controls

specialist, in understanding and selecting the proper choices. Future work may investigate the use of an expert system to guide the user in setting up and running the CFD code.

Selection of the run button on the left, followed by clicking on the run object process button at the lower left, causes the run-time process to be activated. It starts another process that creates the LAPIN input file from the knowledge base records and a template input file containing the names of the input variables. A run-time screen (see fig. 6) for interactively executing and monitoring the simulation is then brought up and the APPL compute process is started, which initiates the LAPIN simulation. (The LAPIN simulation can be run on the same workstation as the ICE operating system or on a remote host, if desired.)

The code execution mode is controlled by the RUN, STOP, and RESET buttons at the lower left of the screen. Code execution and display updates both occur in near-real time. The upper left quadrant shows the code execution status (time and number of time steps) and slider bars for interactively controlling the inlet and exit boundary conditions of free-stream temperature and exit Mach number, respectively. (Not shown is a slider bar for manipulating the position of the inlet's translating centerbody.) In this case, LAPIN is simulating the response of the HSCT-baseline inlet without control to a 2.5% step-increase in ambient temperature. The upper right quadrant shows the inlet's normal-shock-position time history (0-0.2 sec.). The inlet Mach number is initially 2.35 and the exit Mach number is held constant at 0.425 (constant corrected mass flow). The lower left quadrant shows a scrollable table of normal-shock position at every 0.01 second. The lower right plot shows the shock time history for the same perturbation, but with a constant exit volumetric flow rate. The two responses are significantly different and illustrate the importance of having a correct compressor-face boundary condition as was mentioned in the "Computational Fluid Dynamics Activities" section.

Once a simulation run has been completed, the user has the option of saving the CFD results file as a knowledge base record or discarding it. When saved in the knowledge base, it automatically becomes associated with records used to create the LAPIN input file.

Concluding Remarks

The design of propulsion controls for high speed (Mach 2+) vehicles, such as the high speed civil

transport (HSCT), could benefit significantly from modeling based on computational fluid dynamics (CFD) simulations. Unfortunately, CFD simulations are seldom, if ever, used because they: 1) are not interactive and easily operated by non-CFD experts; 2) have relatively long execution times that are impractical for controls problems; 3) can have millions of states for which no suitable/robust model-reduction techniques are available and 4) generally lack moving geometry surfaces. The paper describes a controls/CFD research program in which a team of specialists from the controls, CFD, and computational technologies disciplines are working together to remove these obstacles to CFD use.

Considerable progress has been made in each discipline area. Controls/CFD interface requirements were defined to permit the use of CFD simulation as a "numerical experiment" and to facilitate extraction of information for control analysis and design. The direct use of one-dimensional CFD methods to obtain control models was selected as the initial approach for linear modeling. A major advantage of the method is that only steady-state operating conditions are required from the CFD simulation. The method was extended to include the determination of uncertainty bounds. PARC, an existing multidimensional CFD code, was modified to permit time-accurate operation and specification of boundary conditions as a function of time. Other modifications were made to provide more realistic boundary conditions for bleed regions and at the compressor face. Parallelization of PARC has resulted in speedups that hold promise for practical execution of controls related problems. Use of existing system integration software has facilitated progress toward the development of an integrated analysis/design system. Its features include a knowledge base of information relating to the "numerical experiment", tools for managing the knowledge base, and a variety of user interface services. The software is structured to simplify the development of application-specific services, such as a run-time monitor. Initial capabilities of interactive execution, run-time monitoring, and acquisition of CFD results have been demonstrated for an unsteady simulation of a HSCT-baseline inlet based on the one-dimensional LAPIN code. A message-passing approach permits execution of the CFD simulation to take place on remote distributed/parallel processors.

In the future the controls research may explore methods for extracting reduced-order models from multidimensional, viscous simulations and will be extended to other disciplines such as structures. Planned modifications to the PARC code include the

capability to simulate inlet variable geometry features, such as translating/collapsing centerbodies and overboard bypass doors, and more realistic bleed/bypass and compressor face boundary conditions. Further reduction in execution times using parallel processing techniques will be pursued. Modifications to the integrated analysis design system are planned to permit interactive operation of PARC and processing of CFD results for controls analysis and direct use of controls analysis tools. Future work may also investigate the use of an expert system to guide the user in setting up and running the CFD code. By combining advanced computational technologies with the controls and CFD research activities of the controls/CFD program, routine use of CFD simulations by the controls discipline is coming closer to reality.

References

1. Chicatelli, A., et. al., "Interdisciplinary Modeling Using Computational Fluid Dynamics and Control Theory," Proceedings of the 1994 American Controls Conference, Baltimore, MD, June 29-July 1, 1994.
2. Paynter, Gerald C., "CFD Status for Supersonic Inlet Design Support," AIAA Paper 94-0465, Jan. 1994.
3. Chicatelli, A., "Methods for Developing Linear Reduced Order Models of Internal Flow Propulsion Systems," M.S. Thesis, University of Akron, May 1990.
4. Varner, M.O., et. al., "Large Perturbation Flow Field Analysis and Simulation for Supersonic Inlets," NASA CR-174676, Sept. 1984.
5. Cooper, G.K. and Sirbaugh, J.R., "The PARC Code: Theory and Usage," AEDC-TR-89-15, Dec. 1989.
6. Chung, J., "Numerical Simulation of a Mixed-Compression Supersonic Inlet Flow," AIAA Paper 94-0583, Jan. 1994.
7. CFD Laboratory, Dept. of Mechanical Engineering, "Parallelization of PARC2D Code: Status Report #1," Indiana University Purdue University, Indianapolis, IN, NASA Grant NAG3-1577, Jan. 1994.
8. Quealy, A., Cole, G., and Blech, R.A., "Portable Programming on Parallel/Networked Computers Using the Application Portable Parallel Library (APPL)," NASA TM 106238, July 1993.
9. Akay, H.U., private communication, Dept. of Mechanical Engineering, Indiana University Purdue University, Indianapolis, IN, Apr. 1994.
10. Khandelwal, S.C., private communication, NYMA, Inc., Cleveland, OH, Apr. 1994.
11. Babrauckas, T.L. and Arpasi, D.J., "Integrated CFD and Experiments Real-Time Data Acquisition Development," ASME paper 93-GT-97, May 1993.

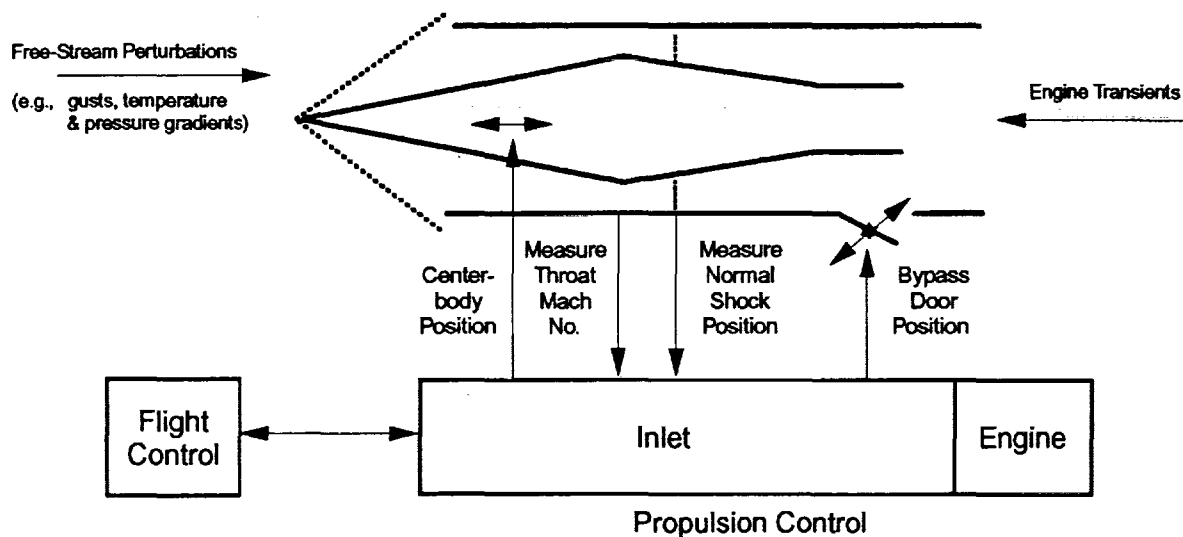


Figure 1 - HSCT Inlet Control System

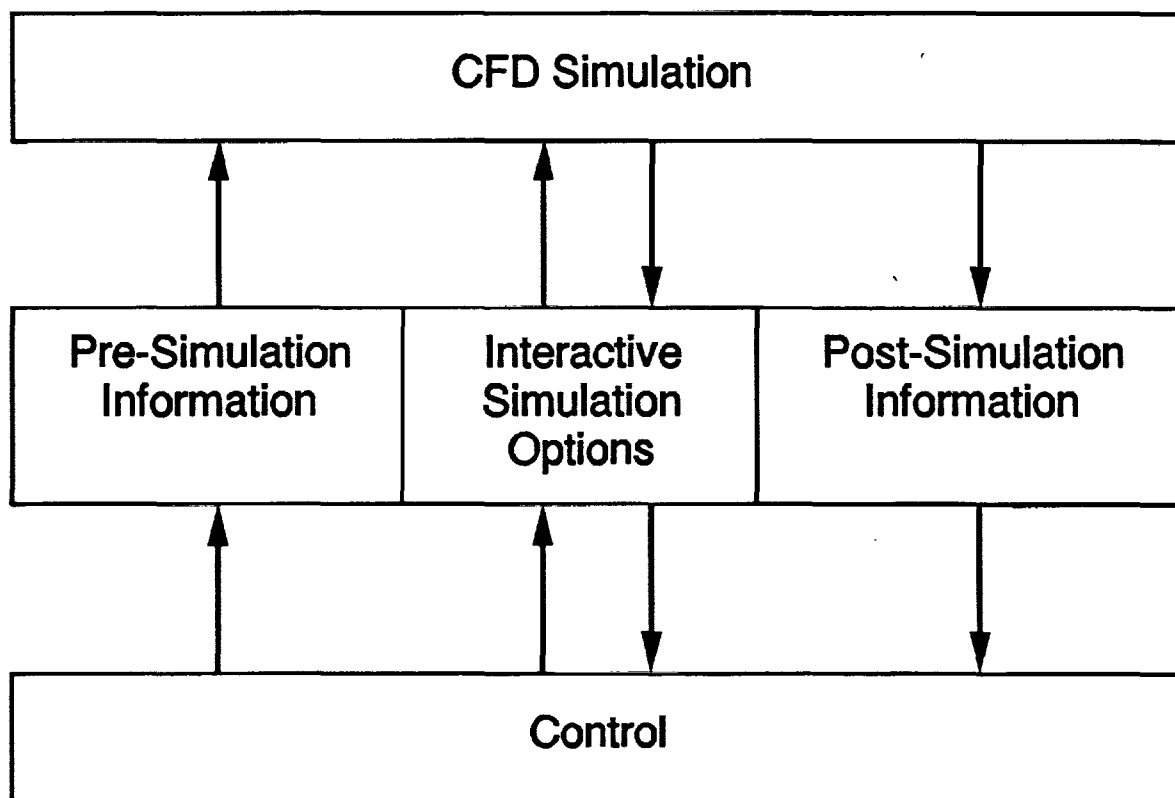


Figure 2 - Controls/CFD Interface Requirements

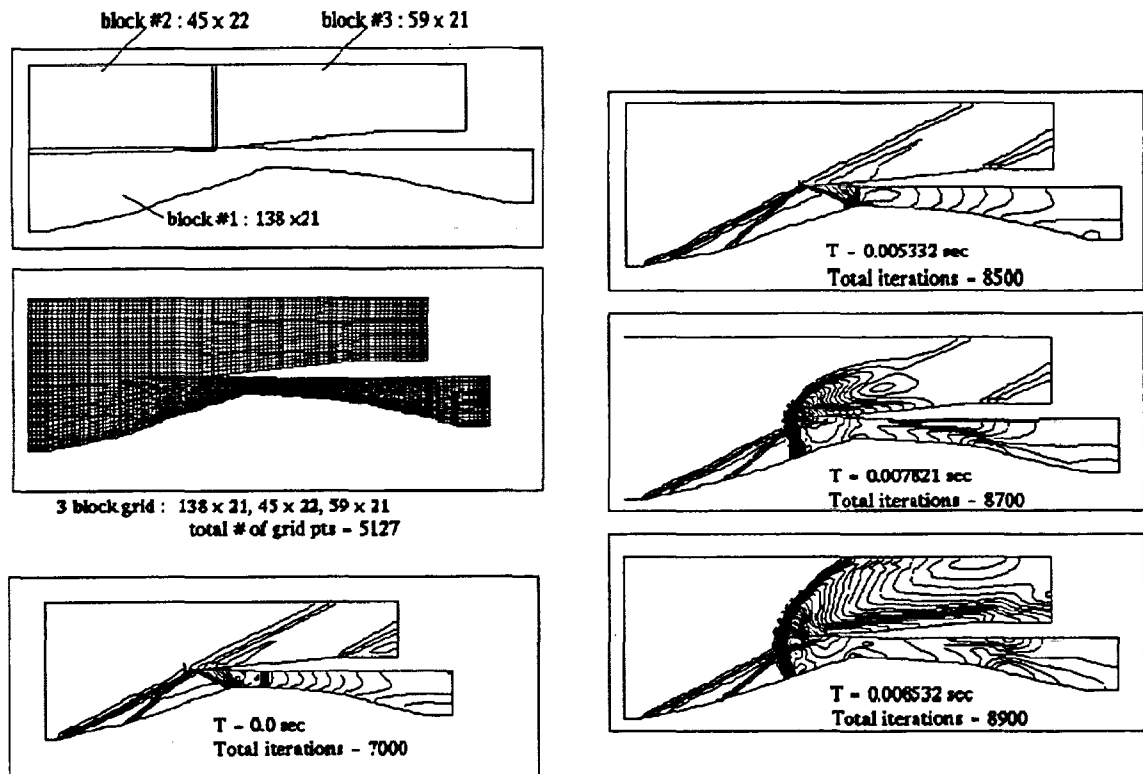


Figure 3 - Inlet unstart transient calculation using inviscid PARC2D. Variable-diameter-centerbody, mixed-compression inlet operating at free-stream conditions of Mach 2.5 and static temperature of 395°R. Compressor-face average Mach number constant at 0.29. Step increase of free-stream temperature of 11.85°R applied at time T=0.0 seconds. Integration time-step size of 3.555×10^{-6} seconds. Total CPU time of 300 seconds on a single Cray Y-MP processor.

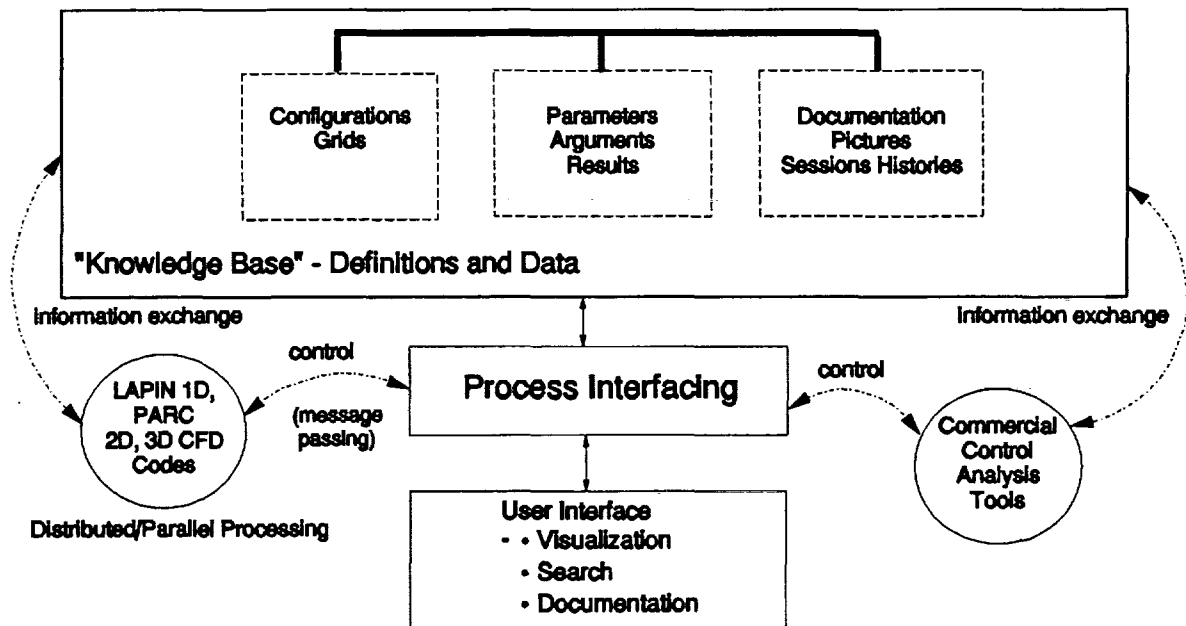


Figure 4 - Integrated Analysis/Design System Features

Large-Perturbation Inlet		NASA LEWIS RESEARCH CENTER		ICE Operating System	
Ice_object_selector: LAPIN knowledge base					
<input type="checkbox"/>	ts[1]	1			
<input type="checkbox"/>	->config[1]	1			
<input type="checkbox"/>	->grid[1]	1			
<input type="checkbox"/>	->bleed[1]	1			
<input type="checkbox"/>	->bypass[1]	1			
<input type="checkbox"/>	->source-term[1]	1			
<input type="checkbox"/>	->free-stream-cond[1]	1			
<input type="checkbox"/>	->BC-spec[1]	1			
<input checked="" type="checkbox"/>	->cfp-param[1]	1			
<input type="checkbox"/>	->acq[1]	1			
<input type="checkbox"/>	->run[1]	1			
<input type="checkbox"/>	->results[1]	1			
<input type="checkbox"/>	->analysis[0]	1			
obj_visual(2): LAPIN/ta.1_cfd-param.1					
<div> <div>04-00</div> <div>PIAT</div> </div>					
<input type="checkbox"/>	id	nominal			
<input type="checkbox"/>	code	LAPIN			
<input type="checkbox"/>	picon	.false.			
<input checked="" type="checkbox"/>	algorithm	Split Characteristics			
<input type="checkbox"/>	ra	0.0000			
<input type="checkbox"/>	nst	2			
<input type="checkbox"/>	ndtc	1			
<input type="checkbox"/>	tiadtc	0.00000			
<input type="checkbox"/>	dtlw	0.00010			
<input type="checkbox"/>	nc	10			
<input type="checkbox"/>	PIX	1			
<div> <div>RUN OBJECT PROCESS</div> <div>LAPIN</div> <div>VIEW</div> <div>EDIT</div> <div>SRCH</div> <div>SHELL</div> </div>					

Figure 5 - Knowledge-base Display for LAPIN CFD Simulation of HSCT-baseline Inlet

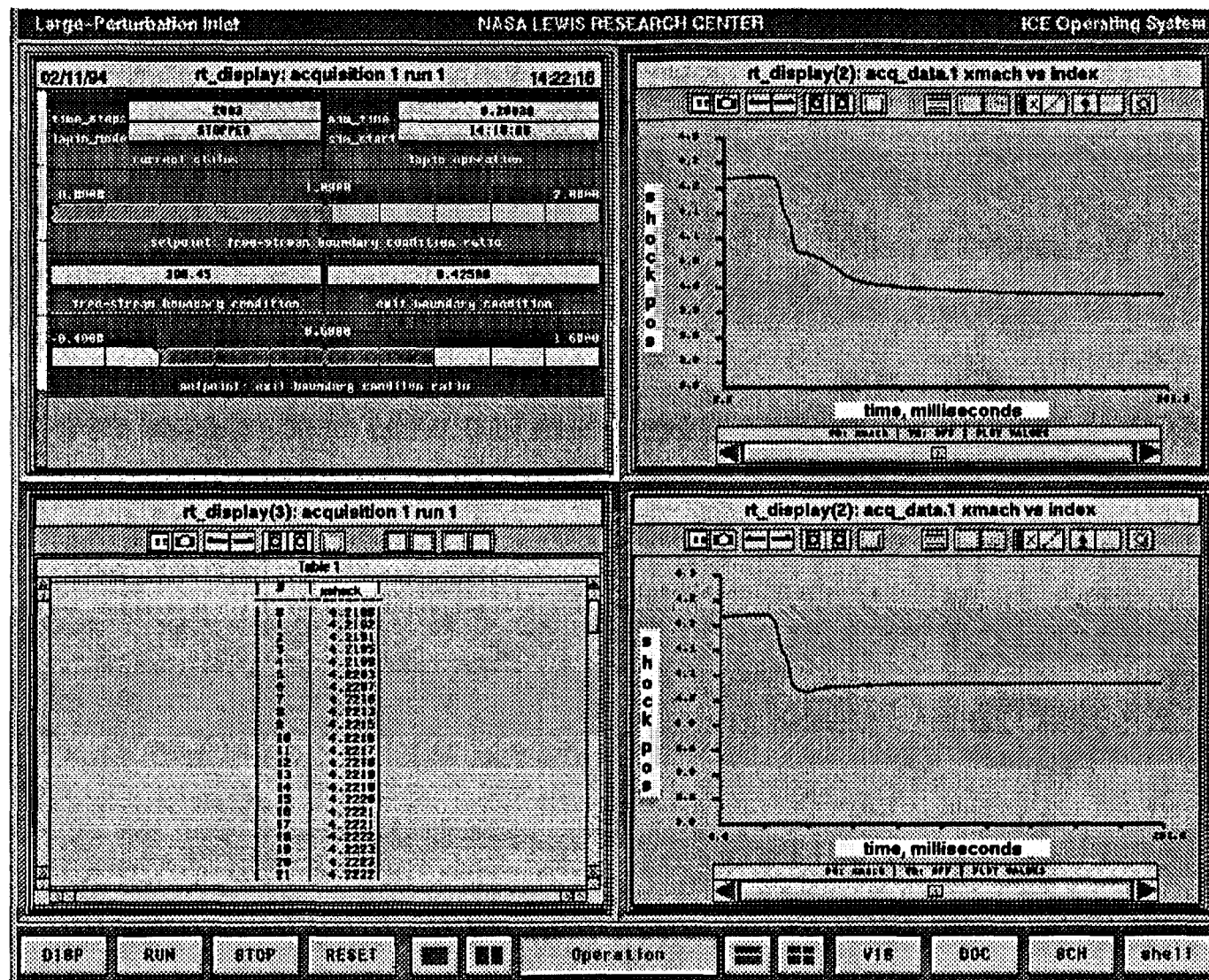


Figure 6 - ICE Operating Panel for Interactive Execution and Monitoring of HSCT Inlet Simulation

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE May 1994		3. REPORT TYPE AND DATES COVERED Technical Memorandum
4. TITLE AND SUBTITLE Computational Methods for HSCT-Inlet Controls/CFD Interdisciplinary Research			5. FUNDING NUMBERS WU-505-62-52	
6. AUTHOR(S) Gary L. Cole, Kevin J. Melcher, Amy K. Chicatelli, Tom T. Hartley, and Joongkee Chung				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-8903	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA TM-106618 AIAA-94-3209 ICOMP-94-10	
11. SUPPLEMENTARY NOTES Prepared for the 30th Joint Propulsion Conference cosponsored by AIAA, ASME, SAE, and ASEE, Indianapolis, Indiana, June 27-29, 1994. Gary L. Cole and Kevin J. Melcher, NASA Lewis Research Center; Joongkee Chung, Institute for Computational Mechanics in Propulsion, NASA Lewis Research Center (work funded by NASA Cooperative Agreement NCC3-233); Amy K. Chicatelli and Tom T. Hartley, University of Akron, Akron, Ohio 44325 (work funded by NASA Grant NAG3-1450). ICOMP Program Director, Louis A. Povinelli, organization code 2600, (216) 433-5818.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 07			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A program aimed at facilitating the use of computational fluid dynamics (CFD) simulations by the controls discipline is presented. The objective is to reduce the development time and cost for propulsion system controls by using CFD simulations to obtain high-fidelity system models for control design and as numerical test beds for control system testing and validation. An interdisciplinary team has been formed to develop analytical and computational tools in three discipline areas — controls, CFD, and computational technology. The controls effort has focused on specifying requirements for an interface between the controls specialist and CFD simulations and a new method for extracting linear, reduced-order control models from CFD simulations. Existing CFD codes are being modified to permit time accurate execution and provide realistic boundary conditions for controls studies. Parallel processing and distributed computing techniques, along with existing system integration software, are being used to reduce CFD execution times and to support the development of an integrated analysis/design system. This paper describes: the initial application for the technology being developed, the high speed civil transport (HSCT) inlet control problem; activities being pursued in each discipline area; and a prototype analysis/design system in place for interactive operation and visualization of a time-accurate HSCT-inlet simulation.				
14. SUBJECT TERMS Interdisciplinary research; Cross-discipline research; Numerical experiment; Parallel processing; Distributed processing; Linear modeling; Controls; HSCT; PARC; CFD			15. NUMBER OF PAGES 13	
			16. PRICE CODE A03	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	